

VCSQuickReference(V1.0,March6th,2001)

GraphicsMethods(x=vcs.init())

Functions		GraphicsMethods(x=vc\$init())	
Canvas,x=vc\$init()		Creating/Retrieving/Deleting	
open/clear/close	()	x.create"graphicsmethod"	(name)'name'or'copyfrom')
plot	(slab,[grid=gad],graphicmethod),[templateobject],...)	x.get"graphicsmethod"	(name)
mode	=1/0	x.removeobject	(object)
update	0		
flush	0		
portrait/landscape()			
page	(
geometry	(width,height,xoffleft,yofftop)resizewindow		
set	(method,'name')		
setcolormap	(name)		
setcolorcell	(icell,r,g,b)		
getcolormap	(name)		
createcolormap	(name)		
getcolormapname	()		
getcolorcell	(icell)		
colormapgui	0		
animate	0		
grid	(dim10,dim11,...,dimn,dimm1)		
for			
resetgrid	()		
printer	(printname,[p,"/T])		
postscript	(filename,[ps],[p,"/T])		
eps	(filename,[ps],[p,"/T])		
gif	(filename,gif,[l,a,"r"],[p,"/T"])		
cgm	(filename,cgm,[l,a,"r"])		
raster	(filename,[ras],[l,a,"r"])		
pslogif	(filename,ps,[l,T,"r"])		
CommonAttributes		CommonAttributes	
all	name	='name'	='name'
	projection	='linear','mollweide','robinson','or'polar'	='linear','mollweide','robinson','or'polar'
	xylabellabels	[1:2]	=list[ordic]
	datawc	[x][1:2]	=list[ordic]
	xylaxisconvett		=valuer#Worldcoordinates
			='linear','log10','ln','exp','or'area_wt'
SpecificAttributes		SpecificAttributes	
boxfill	level_1	=value#Plotdataatthenextlevel_1	=value#Plotdataatthenextlevel_1
	level_2	=value#Plotdataatthenextlevel_2	=value#Plotdataatthenextlevel_2
	color_1	=0<=value<=255#Firstcolorinpalette	=0<=value<=255#Firstcolorinpalette
	color_2	=0<=value<=255#Lastcolorinpalette	=0<=value<=255#Lastcolorinpalette
	legend	=list[,tuple]or'dictionary'	=list[,tuple]or'dictionary'
	ext_1	=h'or'#/Leftarrow(off/on)	=h'or'#/Rightarrow(off/on)
	ext_2	=0<=value<=255#colorofmissingvalue	=0<=value<=255#colorofmissingvalue
isfill	levels	=list[tuple]#ext1[0,20,30]or(20,30),(50,60)]	=list[tuple]#ext1[0,20,30]or(20,30),(50,60)]
	fillareacolors	=list[tuple]0<=values<=255	=list[tuple]0<=values<=255
	fillareastyle	=solid,'hatch','or'pattern'	=solid,'hatch','or'pattern'
	fillareaindices	=1<=value<=17#fillareaab	=1<=value<=17#fillareaab
	ext_1	=h'or'#/Leftarrow(off/on)	=h'or'#/Leftarrow(off/on)
	ext_2	=0<=value<=255#colorofmissingvalue	=0<=value<=255#colorofmissingvalue
missing			
Output		Output	
p/l	portrait/landscape	p/l	portrait/landscape
p/l	portrait/landscape	p/l	portrait/landscape
p/l	portrait/landscape	p/l	portrait/landscape
air:append/replace	p/l:portrait/landscape	air:append/replace	p/l:portrait/landscape
air:append/replace	p/l:portrait/landscape	air:append/replace	p/l:portrait/landscape

Functions

Creating/Retrieving/Deleting	
open/clear/close	<code>x..create "graphicsmethod"</code>
plot	<code>x..get "graphicsmethod"</code>
mode	<code>x..removeobject</code>
update	<code>()</code>
flush	<code>()</code>
portrait/landscape	<code>togglebetweenportrait/landscape</code>
page	<code>(width,height,yoffsetleft,yoffsettop)resizewindow</code>
geometry	<code>(method,'[name]')</code>
set	<code>(name)</code>
setcolormap	<code>(name)</code>
setcolorcell	<code>(icell,r,g,b)</code>
getcolormap	<code>(name)</code>
getcolormapname	<code>(name)</code>
getcolorcell	<code>(icell)</code>
colornogui	<code>()</code>
animate	<code>(dim10,dim11,...,dimn0,dimm1)</code>
grid	<code>setsizedims(./n).values</code>
for	<code>"outofgraphicsmethod"dimensions unadoid()</code>
resetgrid	<code>()</code>
Common Attributes	
all	<code>name projection xly tictables[1][2] xly imutes[1][2] datawc_x y[1][2] xly yaxisconvert</code>
Specific Attributes	
boxfill	<code>level_1 level_2 color_1 color_2 legend ext_1 ext_2 missing</code>
isoffill	<code>levels fillareacolors fillareastyle fillareaendindices ext_1 ext_2 missing</code>
Output	
printer	<code>(printername.[p'"/T])</code>
postscript	<code>(filename).[psl].[p'"/T])</code>
eps	<code>(filename).[psl].[p'"/T])</code>
gif	<code>(filename).[gfl].[a'"/r']([p'"/T])</code>
cgm	<code>(filename).[cgml].[a'"/r'])</code>
raster	<code>(filename).[rasl].[a'"/r'])</code>
psfogi	<code>(filename).[psl].[r'"/T])</code>
ptl	<code>portrait/landscape</code>
pl	<code>portrait/landscape</code>
air:append/replace	<code>p/l;portrait/landscape</code>
air:prepend/replace	<code>p/l;portrait/landscape</code>
air:r:append/replace	<code>p/l;portrait/landscape</code>
air:r:prepend/replace	<code>p/l;portrait/landscape</code>

Template

Querying($\mathbf{y} - \mathbf{v}_0$, init(), miscarandommethod)

Quarrying (α —versus organisational control)

	Value (gm)	Value (im)
list the option for an objective values they resettle	0.00	0.00
return to graphic method type	0.00	0.00
if it's a graphic method, off	0.00	0.00
1/0/*esobility*/ "can be an	0.00	0.00

Secondary Methods

Help(x=ycs.init.O.getisagraphicmethod)

```
print he help for object  
print he help for the es function  
(sobj ect)  
(function )
```

`= 'name'`

```

= solid', 'dash', 'dot', 'dash-dot', 'long-dash'
or 0,1,2,3,4)
= I<=value<=300
= O<=value<=255

= 'name'
= 'dot', 'plus', 'star', 'circle', 'cross', 'diam'
'triangle_up', 'triangle_down', 'triangle_left',
'triangle_right', 'square', 'diamond_fill';
'triangle_up_fill',
'triangle_right_fill', 'square_fill', or < or >
= I<=value<=300
= O<=value<=255

= 'name'
= 'down_fill', 'triangle_left_fill',
'triangle_left_fill',
= I<=value<=300
= O<=value<=255

= 'name'
= 'solid', 'long-dash', 'dot', 'dash-dot', 'long-dash'
or 'hatch', 'on', 'pattern'
= I<=value<=17
= O<=value<=17

```